(19)日本国特許庁（ＪＰ）　(12)特　許　公　報（Ｂ２）　(11)特許番号

第2576385号

(45)発行日　平成９年(1997)１月29日　(24)登録日　平成８年(1996)11月７日

| (51)Int.Cl.[6] | 識別記号 | 庁内整理番号 | ＦＩ | | 技術表示箇所 |
|---|---|---|---|---|---|
| Ｇ０６Ｆ　12/14 | ３２０ | | Ｇ０６Ｆ　12/14 | ３２０Ｂ | |
| 9/06 | ５５０ | | 9/06 | ５５０Ｂ | |

請求項の数４(全 14 頁)

(21)出願番号　特願平5－270718

(22)出願日　平成５年(1993)10月28日

(65)公開番号　特開平7－129473

(43)公開日　平成７年(1995)５月19日

(73)特許権者　000004237
日本電気株式会社
東京都港区芝五丁目７番１号

(72)発明者　小倉　直志
東京都港区芝五丁目７番１号　日本電気株式会社内

(74)代理人　弁理士　京本　直樹　(外２名)

審査官　野仲　松男

(56)参考文献　特開　平４－147353（ＪＰ，Ａ）
特開　昭60－167033（ＪＰ，Ａ）

(54)【発明の名称】　データ保護装置

(57)【特許請求の範囲】

【請求項１】　１以上のデータ系列の暗号化データを記憶し指定されたアドレスに従って出力する外部記憶装置と、外部記憶装置から暗号化データを入力しデータを復号する復号演算装置と、復号演算装置により復号されたデータを入力し処理するＣＰＵと、ＣＰＵから入力されたアドレスを受けて外部記憶装置にアドレスを出力するアドレス制御回路とを備え、

前記復号演算装置は各データ系列の先頭アドレスの暗号化データの復号には初期化パラメータを用いて復号を行い、先頭アドレス以外の暗合化データの復号には直前のアドレスの復号されたデータを基にパラメータを合成して復号を行い、

前記アドレス制御回路は、ＣＰＵより入力したアドレスの不連続を検出する検出手段と、前記アドレス不連続検出により、ＣＰＵへの復号データの入力を保留し、不連続となった新たに指定するアドレスの属するデータ系列の先頭アドレスから当該不連続アドレスまで順次暗号化データの読み出しと復号を行わしめ、当該不連続アドレスのデータの復号完了後前記保留を解除し、ＣＰＵの復号データの入力を再開させる制御手段を有することを特徴とするデータ保護装置。

【請求項２】　前記入力データ系列に対しデータ対データ対応で結合された非線形変換器または線形変換器を有することを特徴とする請求項１記載のデータ保護装置。

【請求項３】　前記制御手段がキャッシュ・コントローラにより構成されることを特徴とする請求項１又は２記載のデータ保護装置およびデータ保護方式。

【請求項４】　前記外部記憶装置へのアドレス信号出力の一部または全部に対して非線形変換または線形変換を

施したデータを該初期化パラメータとする手段をさらに有することを特徴とする請求項１，２，又は３記載のデータ保護装置。

【発明の詳細な説明】

【０００１】

【産業上の利用分野】本発明は、データ保護装置に関し、特にコンピュータの外部記憶装置に記憶された該コンピュータの実行プログラムに対するデータ保護装置に関する。

【０００２】

【従来の技術】従来のコンピュータの外部記憶装置に記憶されたデータの保護装置および保護方式は、図１４に示すように外部データ・バス４と内部データ・バス５の間に復号化回路３４を有している（例えば、特開昭６１－１６８０６１号公報参照）。

【０００３】図１４に示す従来例では、外部記憶装置２にデータを書き込む場合、ＣＰＵ３は暗号鍵回路３６を起動した後、暗号鍵を暗号化回路３５にセットする。これにより、ＣＰＵ３から内部データ・バス５に出力された書き込みデータは暗号化回路３５を通過するたびに暗号化されて外部データ・バス４に出力され外部記憶装置２に入力される。ＣＰＵ３自体のプログラムを書き込む場合には暗号鍵回路３６、暗号化回路３５を停止して暗号化処理を行わないようにする。

【０００４】外部記憶装置から読み出す場合には、暗号鍵回路３６を起動した後、暗号鍵を復号化回路３４にセットする。これにより、外部記憶装置２から外部データ・バス４上に出力された暗号化データは復号化回路３４を通過するたびに元のデータに復号され、内部データ・バス５をとおしてＣＰＵ３に入力される。ＣＰＵ３のプログラムを読み出す場合にはＣＰＵ３は命令フェッチ制御線４０により復号化回路３４の動作を停止し、外部データ・バス４上のデータを、そのまま内部データ・バス５に出力するように制御する。

【０００５】暗号化回路３５、復号化回路３４、暗号鍵回路３６の構成は第三者に対して秘密に保持する必要がある。

【０００６】以上説明したように図１４に示す従来例では、データの暗号化のみでＣＰＵ３自体のプログラムの暗号化を行わない。

【０００７】ＣＰＵ３自体のプログラムの暗号化を行う例を図１５に示す。外部データ・バス４と内部データ・バス５の間に変換テーブル８を有している。

【０００８】変換テーブル８は外部データ・バス４を入力とし、外部データ・バス４の値を非線形変換または線形変換した信号を内部データ・バス５に出力する。変換テーブル８は、外部データ・バス４と同じビット長の非線形演算または線形演算を行う演算器９と任意に設定可能なデータである暗号鍵１０とにより構成することができる。例えば４ビット長の場合には図１６に示すような１対１の非線形変換を行うものとする。非線形演算器９の例としては特開平０３－２５４５３８号公報や特公昭５９－４５２６９号公報がある。暗号鍵１０を定数とすれば変換テーブル８はあらかじめ各入力信号に対応する出力信号を記憶させた記憶装置により構成することもできる。

【０００９】外部記憶装置２には、図７に示すように、あらかじめ図１６の逆変換テーブルにより非線形変換された暗号化データが記憶されているものとする。アドレス、データのビット長は説明のためそれぞれ４ビットとする。以降、「」で挟まれた各データは特に断らない限り２進数を表すものとする。

【００１０】アドレス「００００」の元のデータ「１００１」は暗号化データ「１１０１」に変換される。同様にアドレス「０００１」の元のデータ「１１１０」は暗号化データ「１０００」に変換される。ここでアドレス「００００」とアドレス「００１１」の元のデータはともに「１００１」なので暗号化データもともに「１１０１」で同じになっている。

【００１１】復号時の動作を図１８に示す。

【００１２】外部記憶装置２は、コンピュータ１から読み出し信号６が入力されると外部アドレス・バス２１のアドレスに対応した暗号化データを、外部データ・バス４に出力する。外部データ・バス４の暗号化データは変換テーブル８により非線形変換されて元のデータとなりＣＰＵ３に入力される。変換テーブル８の動作は第三者に対して秘密に保持する必要がある。

【００１３】１対１変換ではなく１対多の変換を行うデータ秘匿方式の従来例は、図１９に示すように変換テーブル８、信号選択回路１１、記憶回路１４、排他的論理和回路７によりフィードバック・ループを構成する方式を有している。

【００１４】変換テーブル８は復号化バス４２上の信号を入力とし、非線形変換または線形変換を行った信号を信号選択回路１１に出力する。信号選択回路１１は変換テーブル８の出力信号と、初期値１２と、制御回路４３からの選択信号１３を入力とし、選択信号１３が入力されている場合は初期値１２を、それ以外の場合は変換テーブル８の出力信号を、記憶回路１４に出力する。初期値１２は暗号化バス４１と同じビット長の定数である。記憶回路１４は、信号選択回路１１の出力信号と、制御回路４３からのラッチ信号４４を入力とし、ラッチ信号４４が入力されている場合は信号選択回路１１の出力信号を、それ以外の場合は直前のラッチ信号４４が入力されていた時に記憶回路１４が出力していた信号を、排他的論理和回路７に出力する。排他的論理和回路７は暗号化バス４１と記憶回路１４からの信号を入力とし、暗号化バス４１と記憶回路１４の出力信号の排他的論理和演算をした演算結果を復号化バス４２に出力する。制御回路４３は復号動作開始時に選択信号１３を信号選択回路

に出力し、暗号化バス４１のデータの各１語を変換する直前ごとにラッチ信号４４を記憶回路１４に出力する。

【００１５】変換テーブル８は、暗号化バス４１と同じビット長の非線形演算または線形演算を行う演算器９と任意に設定可能なデータである暗号鍵１０とにより構成することができる。例えば４ビット長の場合には図２０に示すような１対１の非線形変換を行うものとする。暗号鍵１０を定数とすれば変換テーブル８はあらかじめ各入力信号に対応する出力信号を記憶させた記憶装置により構成することもできる。

【００１６】暗号化バス４１の暗号化データ列の生成手段を図２１に示す。本例では、暗号化バスのビット幅として４ビット、変換テーブルとして図２０の変換テーブルを用い、元のデータ列の各１語に、時系列順に１、２、３とデータ番号を便宜的に割り当てるものとする。初期値１２は定数「１１０１」とする。

【００１７】データ番号１の元のデータ「１００１」を暗号化するときには、初期値１２「１１０１」と排他的論理和演算をし、「０１００」を暗号化データとして得る。データ番号２の元のデータ「１１１０」を暗号化するときには、データ番号１の元のデータ「１００１」を変換テーブルにより変換した値「１１０１」と、データ番号２の元のデータ「１１１０」を排他的論理和演算をし、「００１１」を暗号化データとして得る。同様に、データ番号３の元のデータ「００１１」を暗号化するときには、データ番号２の元のデータ「１１１０」を変換テーブルにより変換した値「１０００」と、データ番号３の元のデータ「００１１」を排他的論理和演算をし、「１０１１」を暗号化データとして得る。

【００１８】つまりｎを自然数としデータ番号ｎの元のデータ、暗号化データをそれぞれ元のデータ（ｎ）、暗号化データ（ｎ）と表し、変換テーブルによる変数をｆで表すと、データ番号ｎの元のデータは次の式により暗号化される。

【００１９】暗号化データ（ｎ）＝元のデータ（ｎ）
　　元のデータ（ｎ）＝暗号化データ（ｎ）　ｘｏｒ　ｆ（元のデータ（ｎ－１））　……（式１）
　　元のデータ（０）＝初期値

式１に示すとおり、暗号化データ（ｎ）を元のデータ（ｎ）に復号するには元のデータ（ｎ－１）が必要だが、元のデータ（ｎ－１）を得るには元のデータ（ｎ－２）が必要となる。つまり暗号化データ（ｎ）を復号するにはｎ以前のすべての元のデータが必要となるため、暗号化バス４１上の暗号化データから元のデータを推測することは非常に困難であり暗号強度は高い。

【００２６】変換テーブル８の内容は第三者に対して秘密に保持する必要がある。初期値１２も秘密に保持することが望ましいが、復号時には元のデータ（１）にしか用いないため、公開しても暗号強度への影響は比較的小さい。したがって初期値１２は外部からディップ・スイッチなどにより設定することも可能である。

【００２７】

【発明が解決しようとする課題】図１４に示す第一の従来例では、プログラムの秘匿化ができないという問題点があった。また、外部データバス４と内部データ・バス５の間に復号化回路３４が挿入されているため、次に説明する図１５に示す第二の従来例と同じ問題点を有する。

【００２８】図１５に示す第二の従来例では、外部データ・バス４と内部データ・バス５の間に変換テーブル８が挿入されているので信号処理に要する時間は変換テーブル８の参照時間が支配的となる。暗号鍵１０を定数と

し、あらかじめ各入出力信号に対する出力信号を記憶させた記憶装置により変換テーブル８を構成し、変換テーブル８を現在最先端の０．５μｍ～０．８μｍルールのＣＭＯＳプロセスを用いて作成した場合でも、参照時間は少なくとも２０ｎｓ以上は必要である。

【００２９】したがって外部記憶装置からの読み出し時間が３０ｎｓ～１００ｎｓ程度と高速な、近年の外部記憶装置２とＣＰＵ３のシステム構成の場合には、変換テーブル８の参照時間２０ｎｓが外部記憶装置２からの読み出し時間に対して無視できなくなり、第二の従来例によりプログラムの秘匿化を行うにはＣＰＵ３にウエイトをかける、または動作周波数を下げなければならず、結果としてコンピュータ１の性能を大きく低下させるという欠点があった。

【００３０】また、外部データ・バス４上の暗号化データと、内部データ・バス５上の元のデータが１対１に対応しているため、プログラムの秘匿化の場合、各命令の使用頻度に差があることを利用すると、第三者が外部データ・バス４上の暗号化データから元のデータを推測することは、比較的容易である。例えば暗号化データ列中で最も出現頻度の大きい暗号化データが、転送命令に対応するだろうと推測することができる。したがって、外部データ・バス４が８ビット幅程度のコンピュータ１の場合、すべてを推測する計算量は非常に小さく暗号強度は低いという問題点があった。

【００３１】図１９に示した第三の従来例の場合には、１対多への変換であり第二の従来例の問題点はない。暗号化バス４１と復号化バス４２の間には排他的論理和回路７が挿入されているのみであり、排他的論理和回路７の遅延時間は現在の０．５μｍ～０．８μｍルールのＣＭＯＳプロセスで構成した場合数ｎｓ程度であり、近年の高速な外部記憶装置２の読み出し時間に対しても無視できる程度である。また、元のデータ（ｎ）を得るために変換テーブル８に暗号化データ（ｎ）を入力する第二の従来例と異なり、第三の従来例では元のデータ（ｎ）を得るために変換テーブル８に入力するのは元のデータ（ｎ－１）である。したがって変換テーブル８の出力信号を参照するのは変換テーブル８に入力してから１読み出し周期後となり、変換テーブル８の遅延時間が外部記憶装置２からの読み出し動作に影響を与えることはなく、第二の従来例のようにＣＰＵ３の性能を低下させることもない。

【００３２】ただし、第三の従来例の場合には、式１に示すとおり、暗号化データ（ｎ）を元のデータ（ｎ）に復号するには元のデータ（ｎ－１）が必要だが、元のデータ（ｎ－１）を得るには元のデータ（ｎ－２）が必要となる。つまり例えば図２２の暗号化データ（７）を復号するには暗号化データ（１）から暗号化データ（６）までのすべての暗号化データが、データ番号の順番で復号されていなければならない。

【００３３】これに対して、ほとんどのプログラムは分岐命令を有しているため、外部記憶装置から読み出されるデータはアドレスの順番で読み出されるとは限らないため、第三の実施例を適用することは非常に困難である。

【００３４】例えば図２２のデータ番号をアドレスとみなし、アドレス４の元のデータがアドレス７への分岐命令の場合、アドレス１、２、３、４と順番に暗号化データを復号した後、アドレス５の暗号化データではなくアドレス７の暗号化データを復号しようとするが、アドレス６の元のデータが記憶回路１４に記憶されていないため正しく復号することができない。

【００３５】したがって第三の実施例は、コンピュータのプログラムの秘匿化に適用することは非常に困難であるという問題点を有する。

【００３６】

【課題を解決するための手段】本発明のデータ保護装置およびデータ保護方式は、１以上のデータ系列の暗号化データを記憶し指定されたアドレスに従って出力する外部記憶装置と、外部記憶装置から暗号化データを入力しデータを復号する復号演算装置と、復号演算装置により復号されたデータを入力し処理するＣＰＵと、ＣＰＵから入力されたアドレスを受けて外部記憶装置にアドレスを出力するアドレス制御回路を備え、前記復号演算装置は各データ系列の先頭アドレスの暗号化データの復号には初期化パラメータを用いて復号を行い、先頭アドレス以外の暗合化データの復号には直前のアドレスの復号されたデータを基にパラメータを合成して復号を行い、前記アドレス制御回路は、ＣＰＵより入力したアドレスの不連続を検出する検出手段と、前記アドレス不連続検出により、ＣＰＵへの復号データの入力を保留し、不連続となった新たに指定するアドレスの属するデータ系列の先頭アドレスから当該不連続アドレスまで順次暗号化データの読み出しと復号を行わしめ、当該不連続アドレスのデータの復号完了後前記保留を解除し、ＣＰＵの復号データの入力を再開させる制御手段を有する。

【００３７】

【実施例】次に本発明について図面を参照して説明する。

【００３８】図１は本発明の第一の実施例の構成を示すブロック図である。

【００３９】変換テーブル８は内部データ・バス５上の信号を入力とし、非線形変換または線形変換を行った信号を信号選択回路１１に出力する。信号選択回路１１は変換テーブル８の出力信号と、初期値１２と、アドレス制御回路１５からの選択信号１３を入力とし、選択信号１３が入力されている場合は初期値１２を、それ以外の場合は変換テーブル８の出力信号を、記憶回路１４に出力する。初期値１２は外部データ・バス４と同じビット長の定数である。記憶回路１４は、信号選択回路１１の

出力信号と、アドレス制御回路１５からの読み出し信号６を入力とし、読み出し信号６が有効のときには信号選択回路１１の出力信号を、読み出し信号６が無効のときには直前の読み出し信号６が有効のときに記憶回路１４が出力していた信号を、排他的論理和回路７に出力する。排他的論理和回路７は外部データ・バス４と記憶回路１４からの信号を入力とし、外部データ・バス４と記憶回路１４の出力信号の排他的論理和演算をした演算結果を内部データ・バス５に出力する。アドレス制御回路１５はＣＰＵ３からの内部アドレス・バス１６と内部読み出し信号１７と分岐信号１８と読み出しクロック信号１９を入力とし、外部記憶回路２に外部アドレス・バス２１と読み出し信号６を出力し、記憶回路１４に読み出し信号６を出力し、ＣＰＵ３に対しＢＵＳＹ信号３を出力する。

【００４０】外部記憶装置２には一つまたは複数のデータ系列が暗号化されてあらかじめ記憶されている。各データ系列長は等しく、ここでは４とする。また各データ系列の先頭データのアドレスのうち、データ系列長に相当する下位ビットは互いに等しいものとする。ここではデータ系列長を４としたので下位２ビットは互いに等しいことになり、ここでは「００」とする。

【００４１】ＣＰＵ３は外部記憶装置２からのデータの１語読み出し動作ごとに内部アドレス・バス１６と内部読み出し信号１７を出力し、分岐命令を実行して内部アドレス・バス１６が不連続になると分岐信号１８を出力し、内部読み出し信号１７に周期も等しく読み出し動作保留中も出力される読み出しクロック信号１９を出力する。また、ＣＰＵ３はＢＵＳＹ信号２０が有効の時には外部記憶装置２に対する読み出し動作を保留する。

【００４２】アドレス制御回路１５は内部アドレス・バス１６と外部アドレス・バスが等しくない期間ＢＵＳＹ信号２０を出力する。

【００４３】分岐信号１８が入力されると、外部記憶装置２に記憶されている暗号データ系列の系列長に相当するビット数の外部アドレス・バス２１の下位ビット数をその暗号データ系列の先頭アドレスの系列長に相当する下位ビットに変更して外部アドレス・バス２１に出力するとともに選択信号１３を出力する。ここでは分岐信号１８が入力されるごとに外部アドレス・バス２１の下位２ビットを「００」に変更する。また、内部読み出し信号１７または、ＢＵＳＹ信号２０が有効の間は読み出しクロック信号１９を読み出し信号６として出力する。これによりＣＰＵ３が読み出し動作を保留している期間も外部記憶装置２からは暗号データ系列が読み出されることになる。

【００４４】変換テーブル８は、外部データ・バス４と同じビット幅の非線形演算または線形演算を行う演算器９と任意に設定可能なデータである暗号鍵１０とにより構成することができる。例えば４ビット幅の場合には図２０に示すような１対１の非線形変換を行うものとする。暗号鍵１０を定数とすれば変換テーブル８はあらかじめ各入力信号に対応する出力信号を記憶させた記憶装置により構成することもできる。

【００４５】変換テーブル８、信号選択回路１１、初期値１２、選択信号１３、記憶回路１４、排他的論理和回路７の動作は第三の従来例と同様である。

【００４６】アドレス制御回路１５の一構成例を図２に、信号波形図を図３に示す。ここでは暗号データ系列の系列長は１６、各系列の先頭アドレスの下位４ビットは「００００」とし、外部アドレス・バス２１のビット幅は１６とする。

【００４７】４ビット・カウンタ２２にはリセット信号として分岐信号１８が、カウント・読み出しクロック信号として読み出し信号６が入力され、そのカウンタ出力信号２７を４ビット・セレクタ２４に出力する。４ビット・セレクタ２４は内部アドレス・バス１６の下位４ビットと、カウンタ出力信号２７とＢＵＳＹ信号２０を入力とし、ＢＵＳＹ信号２０が有効のときにはカウンタ出力信号２７を、ＢＵＳＹ信号２０が無効の時には内部アドレス・バス１６の下位４ビットを外部アドレス・バスの下位４ビットに出力する。比較回路２３は内部アドレス・バス１６の下位４ビットとカウンタ出力信号２７を入力とし、２つの入力信号が等しくない場合ＢＵＳＹ信号を出力する。０判定回路２５はカウンタ出力信号２７を入力とし、入力が０の場合には選択信号１３を出力する。

【００４８】１ビット・セレクタ２６は内部読み出し信号１７と読み出しクロック信号１９とＢＵＳＹ信号２０を入力とし、ＢＵＳＹ信号２０が有効の時には読み出しクロック信号１９を、無効の時には内部読み出し信号１７を外部読み出し信号６に出力する。

【００４９】図３の信号波形図を用いてアドレス制御回路１５の動作を説明する。以降、特に断らない限りＨで終わる数は１６進数を表すものとする。

【００５０】ＣＰＵ３がアドレス１２４２Ｈで分岐命令を実行し、アドレス３４３２Ｈに分岐した場合を例として説明する。

【００５１】アドレス１２４２Ｈまでは順次連続してデータ系列を読み込むため分岐信号１８は入力されず、カウンタ出力信号２７と内部アドレス・バス１６の下位ビットは一致している。アドレス１２４０Ｈになると新しい暗号データ系列に移行するため選択信号１３を出力する。

【００５２】アドレス１２４２Ｈからアドレス３４３２Ｈに分岐するとＣＰＵ３から分岐信号が入力され４ビット・カウンタ２２が０にリセットされ選択信号１３が出力されるとともに、カウンタ出力信号２７が内部アドレス・バス１６の下位４ビットと一致しなくなるためＢＵＳＹ信号２０が発生する。外部アドレス・バス２１は３

４３０Ｈとなり、以後は読み出し信号６により外部アドレス・バス２１は３４３１Ｈ、３４３２Ｈと変化していく。このとき外部記憶装置２から暗号化データ系列が読み出されて行くが、３４３０Ｈ、３４３１ＨのときにはＢＵＳＹ信号２０がＣＰＵ３に入力されているため内部データ・バス５の値は読み込まれない。以降外部アドレス・バス２１が３４３２Ｈとなるとカウンタ出力信号２７と内部アドレス・バス１６の下位４ビットが一致するためＢＵＳＹ信号２０が無効となりＣＰＵは読み出し動作を再開する。

【００５３】ここで、アドレス３４３２Ｈでなくアドレス３４３０Ｈに分岐する場合には、カウンタ出力信号２７と内部アドレス・バス１６の下位４ビットが一致するためにＢＵＳＹ信号２０は発生せず、選択信号１３のみが出力されることになる。

【００５４】暗号化データ系列の生成手順を図４に示す。図４の例では、アドレス、データのビット幅はともに４ビットとする。変換テーブルは図５の変換テーブルを用いる。初期値１２は定数「１１０１」とする。また、暗号データ系列の系列長は４、各系列の先頭アドレスの下位２ビットは「００」とする。

【００５５】図４のとおり、アドレスの下位２ビットが「００」になるたびに元のデータ系列と排他的論理和をとるデータを初期値「１１０１」に変更することのみが第三の従来例と異なり、系列長が４の暗号データ系列を生成する動作は従来例と同様である。

【００５６】本発明の第一の実施例の分岐命令を実行しない場合の動作を図６～図８を用いて説明する。図６は分岐命令を実行しない場合の信号波形図であり、図８はこのときのデータの流れを示す図である。

【００５７】暗号化データ系列の生成時の動作の説明の場合と同様、アドレス、データのビット幅はともに４ビットとする。変換テーブルとして図７の変換テーブルを用いる。初期値１２は定数「１１０１」とする。また、暗号データ系列の系列長は４、各系列の先頭アドレスの下位２ビットは「００」とする。

【００５８】分岐命令を実行しないため分岐信号１８は発生せず、ＣＰＵ３が出力する内部アドレス・バス１６とアドレス制御回路１５が出力する外部アドレス・バス２１の内容は常に一致している。このためアドレス制御回路１５はＢＵＳＹ信号２０を出力せず、ＣＰＵ３は読み出し動作を保留せず内部データ・バス５上の元のデータ系列を内部読み出し信号１７に応じて読み込んでいる。外部アドレス・バス２１の下位２ビットが「００」になるごとに選択信号１３を信号選択回路１３に出力し、内部データ・バス５と排他的論理和をとるデータを初期値「１１０１」に変更する。

【００５９】図８より系列長４の各データ系列の中のデータの流れは第三の従来例と同様であり、各暗号データ系列がすべて正しく元のデータに復号されることが分かる。

【００６０】本発明の第一の実施例の分岐命令を伴う場合の動作を図９～図１１を用いて説明する。図９は分岐命令を伴う場合の信号波形図であり、図１１はこのときのデータの流れを示す図である。ここでは、アドレス「００００」からアドレス「００１０」まで順次外部記憶装置２から読み出し、アドレス「００１０」で分岐命令を実行する結果アドレス「０１１０」に分岐する場合を例に説明する。

【００６１】暗号化データ系列の生成時の動作の説明の場合と同様、アドレス、データのビット幅はともに４ビットとする。変換テーブルとして図１０の変換テーブルを用い、初期値１２は定数「１１０１」とする。また、暗号データ系列の系列長は４、各系列の先頭アドレスの下位２ビットは「００」とする。

【００６２】アドレス「００００」からアドレス「００１０」までは分岐命令を伴わずに順次読み出すため、分岐信号１８が発生せずＣＰＵ３が出力する内部アドレス・バス１６とアドレス制御回路１５が出力する外部アドレス・バス２１の内容は常に一致している。図１１からこの場合のデータの流れは第三の実施例と同じであり、暗号化データが元のデータに正しく復号されていることが分かる。

【００６３】ＣＰＵ３は内部データ・バス５からアドレス「００１０」の元のデータを読み込んだ後アドレス「０１１０」への分岐命令を実行するため、分岐信号１８を出力し、内部アドレス・バス１６に「０１１０」を出力し、外部記憶装置２からアドレス「０１１０」の暗号化データを読み出そうとする。しかし、分岐信号１８により外部アドレス・バス２１の下位２ビットは「００」となるため、アドレス制御回路１５は外部アドレス・バス２１に「０１００」を出力するとともに信号選択回路１１に選択信号１３を出力する。また内部アドレス・バス１６と外部アドレス・バス２１が一致しないのでアドレス制御回路１５がＣＰＵ３にＢＵＳＹ信号２０を出力するため、ＣＰＵ３は読み出し動作を保留する。

【００６４】選択信号１３によって記憶回路１４の記憶データが演算器８の出力ではなく初期値１２となっているため、下位２ビットが「００」から始まる新しい暗号データ系列の先頭データであるところのアドレス「０１００」の暗号化データは排他的論理和回路７により正しく元のデータに復号され内部データ・バスに出力される。

【００６５】この後、ＣＰＵ３は読み出し動作を保留状態だが、アドレス制御回路１５は読み出しクロック信号１９に従って外部アドレス・バス２１に「０１０１」と読み出し信号４を出力し、外部記憶装置２からの読み出し動作を続行する。

【００６６】ＣＰＵ３が読み出し動作を保留している間に内部データ・バス７上に出力されたアドレス「０１０

０」の元のデータと、アドレス「０１０１」の元のデータは、ＣＰＵ３の動作に影響を与えることはなく、次の暗号化データを復号するために演算器８に入力されるだけである。

【００６７】アドレス制御回路１５が読み出しクロック信号１９に従い外部アドレス・バス２１に「０１１０」を出力すると、外部アドレス・バス２１と内部アドレス・バス１６が一致するためＢＵＳＹ信号２０が無効となりＣＰＵ３は読み出し動作を再開する。

【００６８】ここで分岐命令による分岐先アドレス「０１１０」の暗号化データは、すでにアドレス「０１０１」の暗号化データが元のデータに復号されているために、第三の従来例の場合と異なり、正しく元のデータに復号されて内部データ・バス５に出力され、読み出し動作を再開したＣＰＵ３に読み込まれる。

【００６９】以降の動作は次の分岐命令を実行するまでの間、図６～図８で前述した動作と同様である。

【００７０】以上説明した通り、本発明のデータ保護回路は分岐処理を伴う場合でも、従来例と異なり、暗号化データ系列を元のデータ系列に正しく復号することができる。

【００７１】図１２は本発明の第二の実施例の構成を示すブロック図である。

【００７２】本実施例では、アドレス制御回路１５の替わりにキャッシュ・コントローラ２８とキャッシュ・メモリ２９とＣＰＵデータ・バス３０を有しており、その他は、第一の実施例と同様である。キャッシュ・コントローラ２８はフェッチ・バイパス・リプレース動作を行わないものとする。

【００７３】外部記憶装置２には一つまたは複数のデータ系列が暗号化されてあらかじめ記憶されている。各データ系列長は等しく、また各系列の先頭アドレスはデータ系列長に相当する下位ビット数だけ等しい。

【００７４】キャッシュ・コントローラ２８は、ＣＰＵ３からの内部アドレス・バス１６、内部読み出し信号１７、分岐信号１８、読み出しクロック信号１９を入力とし、外部アドレス・バス２１と読み出し信号６とＣＰＵデータ・バス３０と選択信号１３を出力する。またキャッシュ・メモリ２９に対して任意に読み書きが可能である。ＣＰＵ３が読み出し動作を行い内部アドレス・バス１６に読み出しアドレスを出力すると、キャッシュ・コントローラ２８はその読み出しアドレスの元のデータがキャッシュ・メモリ２９に記憶されているかどうかを判別し、記憶されている場合には該当する元のデータをＣＰＵデータ・バス３０に出力する。キャッシュ・メモリ２９に記憶されていない場合、キャッシュ・コントローラ２８は選択信号１３を出力するとともに外部記憶装置２１から該当アドレスを含む暗号化データ系列をその系列の先頭アドレスからその系列の最終アドレスまで読み出す動作を行い、その間ＢＵＳＹ信号２０を出力してＣＰＵ３の読み出し動作を停止させる。このとき外部記憶装置２から読み出された暗号化データ系列は第一の従来例と同様に正しく元のデータ系列に復号されて内部データ・バス５に出力され、キャッシュ・コントローラ２８によって順次キャッシュ・メモリ２９に書き込まれる。キャッシュ・コントローラ２８は該当アドレスを含む暗号化データ系列をその最終アドレスまで復号してキャッシュ・メモリ２９に書き込むと、ＢＵＳＹ信号２０を無効にしてＣＰＵ３の読み出し動作を再開させる。このときは内部アドレス・バス１６上のアドレスに対応する元のデータがキャッシュ・メモリ２９に記憶されているので、キャッシュ・コントローラ２８は該当データをキャッシュ・メモリから読み出し、ＣＰＵデータ・バス３０に出力する。

【００７５】以上説明したように、キャッシュ・メモリ２９を有するコンピュータ１では、キャッシュ・コントローラ２８の動作を活用することによりアドレス制御回路１５を特に有さなくとも本発明のデータ保護装置を構成することができる。

【００７６】図１３は本発明の第三の実施例の構成を示すブロック図である。

【００７７】本実施例は、初期値１２の替わりに第２の変換テーブル３１を有する他は、第一の実施例と同様である。

【００７８】第２の変換テーブル３１は外部アドレス・バス２１上の信号を入力とし、非線形変換または線形変換を行った信号を信号選択回路１１に出力する。

【００７９】第２の変換テーブル８は、外部データ・バス４と同じビット幅の出力を有する非線形演算または線形演算を行う第２の演算器３２と任意に設定可能なデータである第２の暗号鍵３３とにより構成することができる。第２の暗号鍵３３を定数とすれば第２の変換テーブル３１はあらかじめ各入力信号に対応する出力信号を記憶させた記憶装置により構成することもできる。

【００８０】外部記憶装置２に記憶された各暗号化データ系列の先頭データは、その先頭データのアドレスを第２の変換テーブル３１に入力したときの出力信号により暗号化されているものとする。

【００８１】各暗号化データ系列の先頭アドレスはそれぞれ異なるため、各暗号化データ系列の先頭データはそれぞれ異なる信号により暗号化されることになる。

【００８２】第三の実施例の動作は、信号選択回路１１が選択信号１３が入力されている場合に出力する信号が、初期値１２ではなく、外部アドレス・バス２１を入力とした第２の変換テーブル３１の出力信号であることが異なる他は、第一の実施例と同様である。

【００８３】第一の実施例、第二の実施例では、各暗号化データ系列の先頭データは一定の初期値１２と排他的論理和演算をとることにより暗号化、復号化されていたため、各暗号化データ系列の先頭データの暗号化データ

と元のデータの間には従来例２と同様に１対１対応が存在する。このため多数の暗号化データ系列を有する外部記憶装置２の場合には暗号強度が低下する一因になり得る。

【００８４】しかし、第三の実施例では各暗号化データ系列の先頭データは、それぞれ異なる先頭アドレスを入力とした第２の変換テーブル３１の出力信号で排他的論理和されるため、各暗号化データ系列の先頭データの暗号化データと元のデータの間には１対１対応がなく、１対多対応となり、第一の実施例、第二の実施例に比べて暗号強度を向上させることができる。

【００８５】

【発明の効果】本発明による効用として次のものが挙げられる。

【００８６】第一に、本発明の解読のための計算量と、第一の従来例および第二の従来例の解読のための計算量の比率は少なくとも次の値以上となる。

【００８７】従来例に対する解読計算量の比率＝（２の（外部データ・バス４のビット幅）乗）の（暗号化データ系列長－１）乗

例えば外部データ・バス４のビット幅が８で、暗号化データ系列長が４の場合、暗号強度は$10^7$倍以上となり実用上解読される心配はない。

【００８８】第二に、本発明では、外部データ・バス４と内部データ・バス５の間には排他的論理和回路７を一段を挿入するのみであり、外部データ・バス４と内部データ・バス５の間の遅延は無視できるほど小さく、動作クロックが５０ＭＨｚ以上の高速なＣＰＵ３に適用することも容易である。

【００８９】従来例では、外部データ・バス４と内部データ・バス５の間に変換テーブル８が挿入されていたため遅延時間が非常に大きく、せいぜい動作クロックが５ＭＨｚ程度のＣＰＵ３にしか適用できなかった。

【００９０】第三に、本発明のデータ保護装置の回路規模は、従来例と比べてほとんど増加していない。暗号鍵１０、第２の暗号鍵３３を定数としてあらかじめ変換テーブル８、第２の変換テーブル３１を用意すれば、８ビット・データ・バスの場合、５１２バイトのＲＯＭとおよそ１０００ゲート程度で構成できる。

【００９１】暗号強度の観点から比較すると、本発明と同程度の暗号強度を得るために、第２の従来例の変換テーブル８に特開平０３－２５４５３８号公報または特公昭５９－４５２６９号公報を用いた場合には、構成に１０万ゲート以上が必要である。

【００９２】以上説明したとおり、本発明を用いれば外部記憶装置の安全かつ高速なデータ保護装置を安価に構成することができる。

【図面の簡単な説明】

【図１】本発明の一実施例の構成を示すブロック図。

【図２】図１に示すアドレス制御回路のブロック図。

【図３】図２に示すアドレス制御回路の動作を説明する信号波形図。

【図４】図１に示す外部記憶装置に記憶されるデータの暗号処理の動作を説明する流れ図。

【図５】図４で示した変換テーブルの一例。

【図６】図１に示す実施例の分岐命令を実行しない場合の動作を説明する信号波形図。

【図７】図６の説明で用いる図１に示す変換テーブルの一例。

【図８】図６の説明を補足するための流れ図。

【図９】図１に示す実施例の分岐命令を実行した場合の動作を説明する信号波形図。

【図１０】図９の説明で用いる図１に示す変換テーブルの一例。

【図１１】図９の説明を補足するための流れ図。

【図１２】本発明の第二の実施例の構成を示すブロック図。

【図１３】本発明の第三の実施例の構成を示すブロック図。

【図１４】従来例の構成を示すブロック図。

【図１５】第二の従来例の構成を示すブロック図。

【図１６】図１５の説明で用いる図１５に示す変換テーブルの一例。

【図１７】図１５に示す外部記憶装置に記憶されているデータの暗号処理の動作を説明する流れ図。

【図１８】図１５の動作を説明する流れ図。

【図１９】第三の従来例の構成を示すブロック図。

【図２０】図１９の説明に用いる図１１ａに示す変換テーブルの一例。

【図２１】図１９に示す暗号化バス上の暗号化データの作成手順を示す流れ図。

【図２２】図１９の動作を説明する流れ図。

【符号の説明】

１　コンピュータ
２　外部記憶装置
３　ＣＰＵ
４　外部データ・バス
５　内部データ・バス
６　読み出し信号
７　排他的論理和回路
８　変換テーブル
９　演算器
１０　暗号鍵
１１　信号選択回路
１２　初期値
１３　選択信号
１４　記憶回路
１５　アドレス制御回路
１６　内部アドレス・バス
１７　内部読み出し信号

17

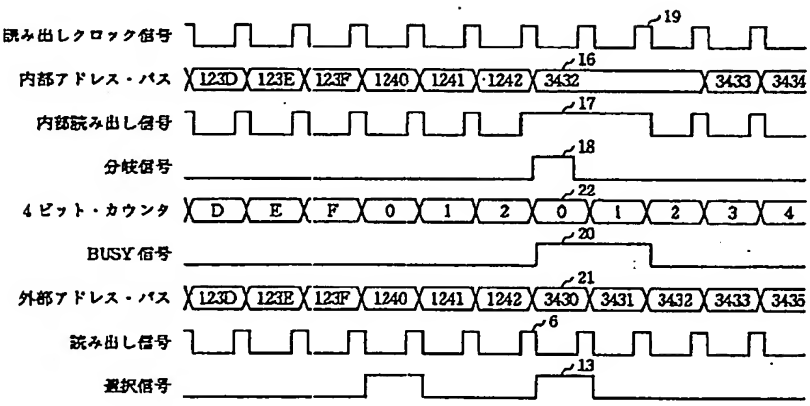
| | |
|---|---|
| １８ | 分岐信号 |
| １９ | 読み出しクロック信号 |
| ２０ | ＢＵＳＹ信号 |
| ２１ | 外部アドレス・バス |
| ２２ | ４ビット・カウンタ |
| ２３ | 比較回路 |
| ２４ | １ビット・セレクタ |
| ２５ | ０判定回路 |
| ２６ | １ビット・セレクタ |
| ２７ | カウンタ出力信号 |
| ２８ | キャッシュ・コントローラ |
| ２９ | キャッシュ・メモリ |
| ３０ | ＣＰＵデータ・バス |
| ３１ | 第二の変換テーブル |

18

| | |
|---|---|
| ＊３２ | 第二の演算器 |
| ３３ | 第二の暗号鍵 |
| ３４ | 復号化回路 |
| ３５ | 暗号化回路 |
| ３６ | 暗号鍵回路 |
| ３７ | 暗号鍵回路制御線 |
| ３８ | 暗号化回路起動線 |
| ３９ | 復号化回路起動線 |
| ４０ | 命令フェッチ制御線 |
| ４１ | 暗号化バス |
| ４２ | 復号化バス |
| １３ | 制御回路 |
| ４４ | トリガ信号 |

＊

【図３】




【図５】

変換テーブル

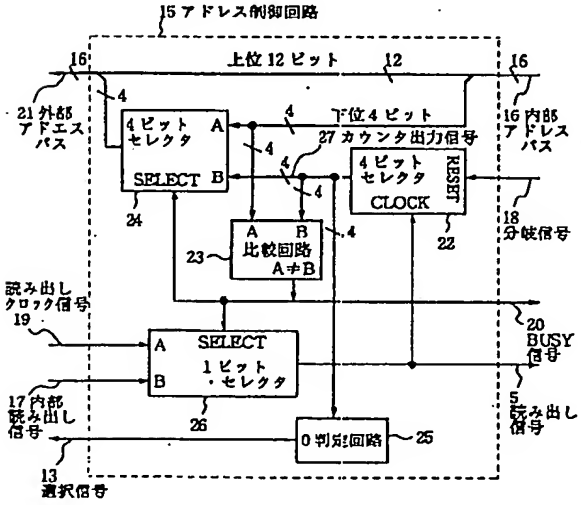
| | |
|---|---|
| 0000→1010 | 1000→1011 |
| 0001→0101 | 1001→1101 |
| 0010→0011 | 1010→0100 |
| 0011→0111 | 1011→1001 |
| 0100→0010 | 1100→0000 |
| 0101→1110 | 1101→0110 |
| 0110→1100 | 1110→1000 |
| 0111→1111 | 1111→0001 |

【図１０】

変換テーブル

| | |
|---|---|
| 0000→1010 | 1000→1011 |
| 0001→0101 | 1001→1101 |
| 0010→0011 | 1010→0100 |
| 0011→0111 | 1011→1001 |
| 0100→0010 | 1100→0000 |
| 0101→1110 | 1101→0110 |
| 0110→1100 | 1110→1000 |
| 0111→1111 | 1111→0001 |

【図２】




【図７】

変換テーブル

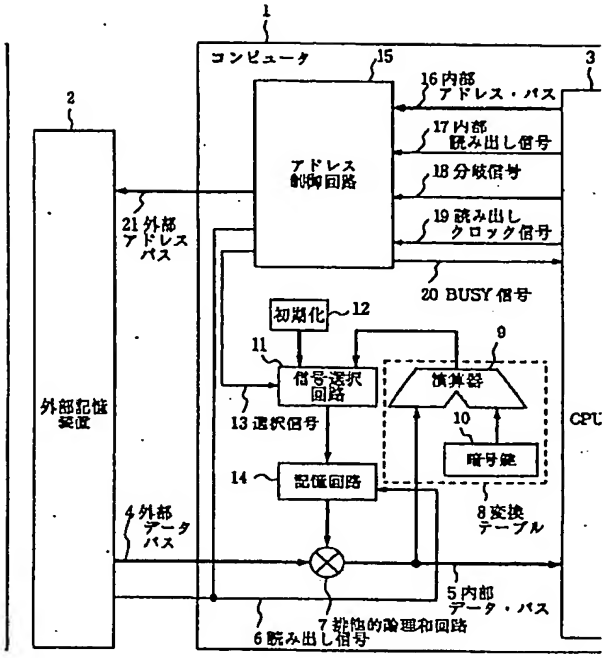
| | |
|---|---|
| 0000→1010 | 1000→1011 |
| 0001→0101 | 1001→1101 |
| 0010→0011 | 1010→0100 |
| 0011→0111 | 1011→1001 |
| 0100→0010 | 1100→0000 |
| 0101→1110 | 1101→0110 |
| 0110→1100 | 1110→1000 |
| 0111→1111 | 1111→0001 |

【図２０】

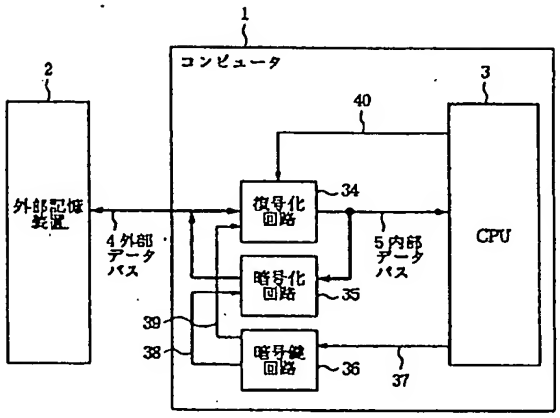
変換テーブル

| | |
|---|---|
| 0000→1010 | 1000→1011 |
| 0001→0101 | 1001→1101 |
| 0010→0011 | 1010→0100 |
| 0011→0111 | 1011→1001 |
| 0100→0010 | 1100→0000 |
| 0101→1110 | 1101→0110 |
| 0110→1100 | 1110→1000 |
| 0111→1111 | 1111→0001 |

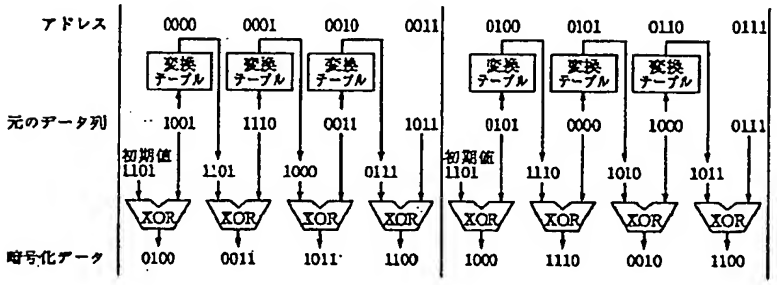
【図１】



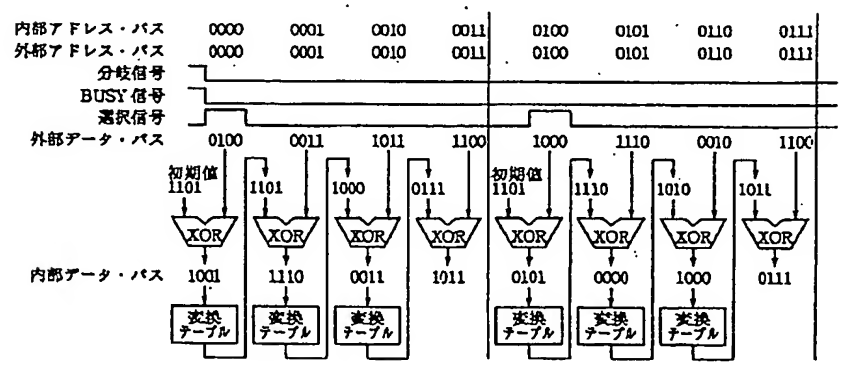

【図１４】



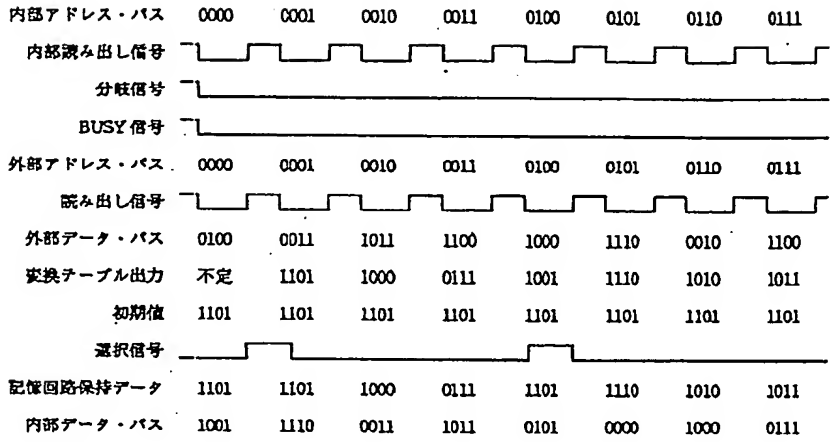

【図４】



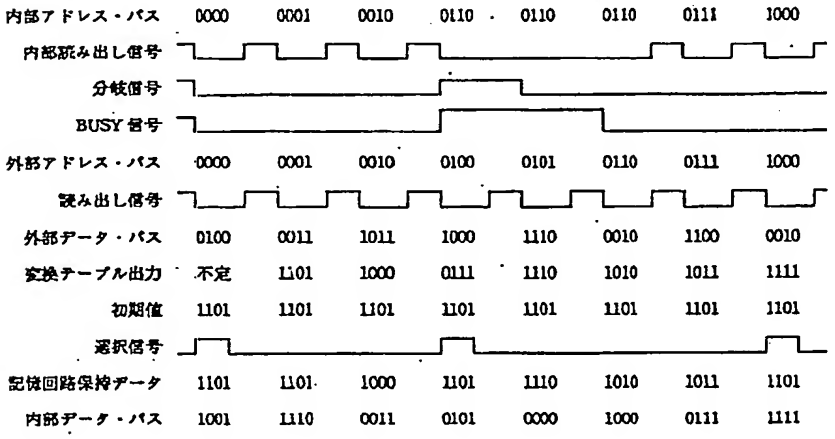

【図８】



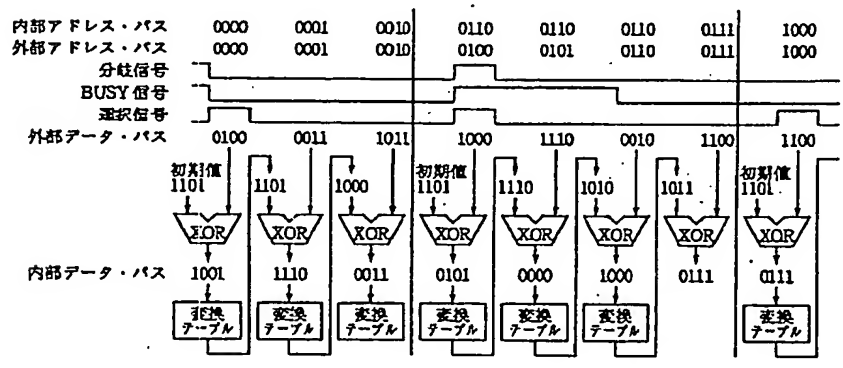
【図６】



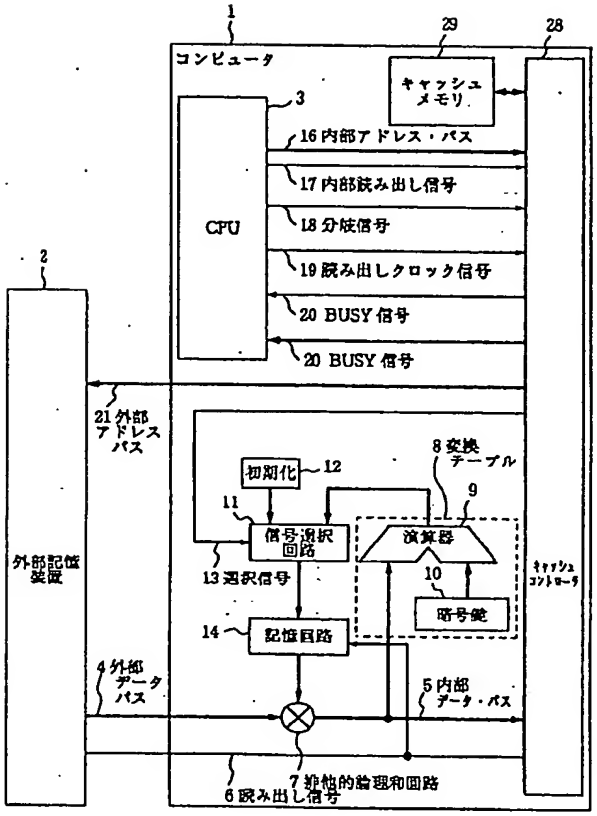

【図９】



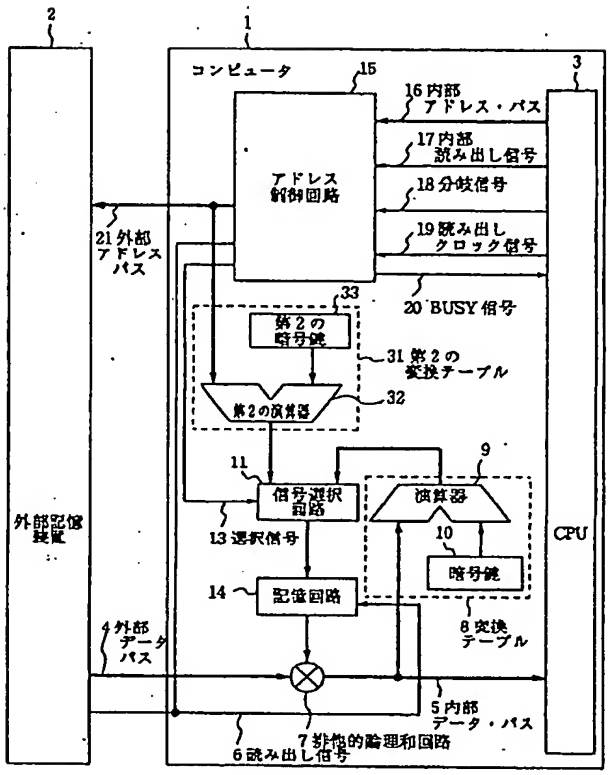

【図１１】



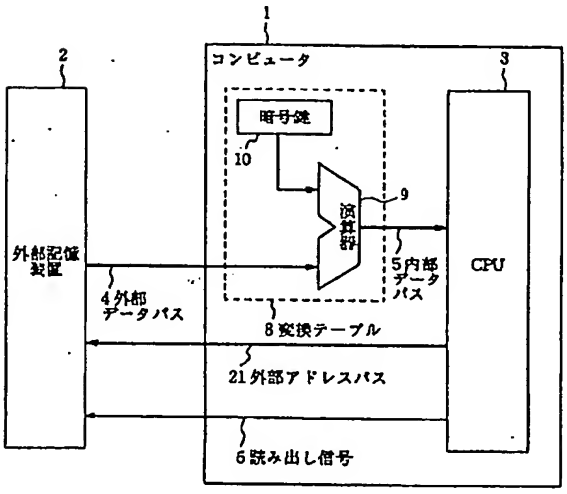
【図１２】




【図１３】




【図１５】




【図１６】

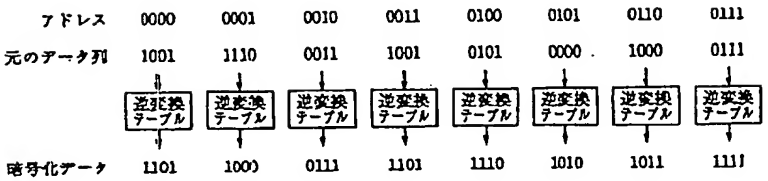
変換テーブル

| | |
|---|---|
| 0000→1010 | 1000→1011 |
| 0001→0101 | 1001→1101 |
| 0010→0011 | 1010→0100 |
| 0011→0111 | 1011→1001 |
| 0100→0010 | 1100→0000 |
| 0101→1110 | 1101→0110 |
| 0110→1100 | 1110→1000 |
| 0111→1111 | 1111→0001 |

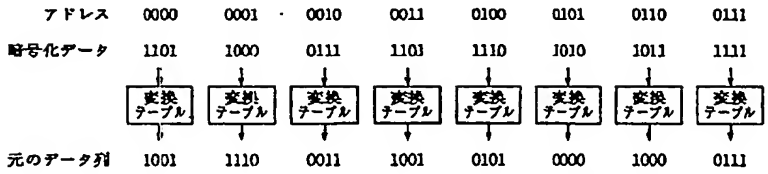
逆変換テーブル

| | |
|---|---|
| 0000→1100 | 1000→1101 |
| 0001→1111 | 1001→1011 |
| 0010→0100 | 1010→0000 |
| 0011→1001 | 1011→1000 |
| 0100→1010 | 1100→0110 |
| 0101→0001 | 1101→1001 |
| 0110→1101 | 1110→0101 |
| 0111→0011 | 1111→0111 |

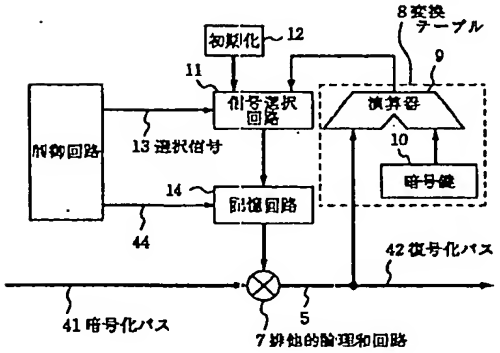
【図１７】



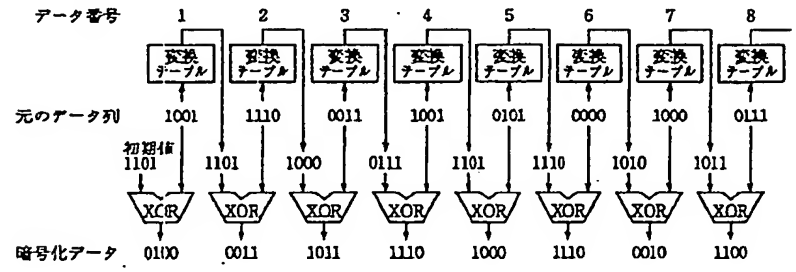

【図１８】




【図１９】




【図２１】



【図２２】